

ALGORITHMS FOR COMPUTING REACHABLE SETS AND CONTROL SETS

Fritz Colonius* Dietmar Szolnoki*

* *Institute for Mathematics, University of Augsburg,
86135 Augsburg, Germany, colonius@math.uni-augsburg.de,
szolnoki@math.uni-augsburg.de*

Abstract: Recently, considerable progress has been made in the numerical computation of reachable sets and control sets. It is the purpose of this paper to survey some of these developments. In general, computation of reachable sets and control sets is a very difficult problem, since the objects which are to be computed have full dimension in the state space. It turns out that certain reformulated problems which in many cases give the desired objects are numerically easier to handle. *Copyright © 2001 IFAC*

Keywords: reachable states, controllability, numerical methods

1. INTRODUCTION

In Sections 2-6, four algorithms are discussed and their relative merits and deficiencies are described. They are based on different reformulations of the problem to compute reachable sets. The continuation algorithm with data reduction computes approximately invariant sets. The box continuation algorithm computes relative chain reachable sets and relative chain control sets; and subdivision algorithms compute viability kernels and hence relative chain control sets and their domains of attraction; furthermore, by computing strongly connected components of symbolic images one obtains relative chain control sets. In analyzing a concrete system, one often has to combine these algorithms in order to take advantage of their relative strengths. There are a number of other methods (not discussed here) to compute reachable sets. In (Colonius and Kliemann, 2000, Appendix C), the computation via time optimal control and Hamilton-Jacobi-Bellman theory is sketched; see also (Falcone *et al.*, 2000). (Camilli *et al.*, 2000) generalize the classical Zubov method for constructing Lyapunov functions to control systems; the viscosity solution of Zubov's equation is a Lyapunov function and can be computed numerically. See also (Grüne, 2001, Chapter 7)

for a discussion, with particular emphasis on the numerical aspects. Ellipsoidal approximations of reachable sets are a powerful tool, see (Kurzanski and Valyi, 1998). However, it seems that there are few results for nonlinear control problems. For this approach and the face lifting approach by (Dang and Maler, 1998) see also the other contributions in this session.

We consider control affine systems of the form

$$\begin{aligned} \dot{x} &= f_0(x) + \sum_{i=1}^m u_i(t) f_i(x), \\ u &\in \mathcal{U} := \left\{ u : \mathbb{R} \rightarrow \mathbb{R}^m, u(t) = (u_i(t)) \in U \right. \\ &\quad \left. \text{for almost all } t \in \mathbb{R} \right\}, \end{aligned} \quad (1.1)$$

with smooth vector fields f_1, \dots, f_m ; the control functions u take values in a given convex and compact subset U of \mathbb{R}^m with $0 \in \text{int } U$. We assume that for every initial state $x \in \mathbb{R}^d$ and every control function $u \in \mathcal{U}$ there exists a unique trajectory $\varphi(t, x, u)$, $t \in \mathbb{R}$. Our primary concern is the computation of reachable sets (orbits) defined as

$$\mathcal{O}^+(x) = \{y \in \mathbb{R}^d, y = \varphi(t, x, u), t > 0, u \in \mathcal{U}\}$$

and of control sets, i.e., subsets of complete controllability:

Definition 1. A subset D of \mathbb{R}^d with nonvoid interior is a control set if $D \subset \text{cl}\mathcal{O}^+(x)$ for all $x \in D$ and D is maximal with this property.

We assume local accessibility, i.e., for all $x \in \mathbb{R}^d$ and all $T > 0$ one has

$$\text{int}\mathcal{O}_{\leq T}^+(x) \neq \emptyset \text{ and } \text{int}\mathcal{O}_{\leq T}^-(x) \neq \emptyset,$$

where $\mathcal{O}_{\leq T}^-(x) = \{y \in \mathbb{R}^d, x = \varphi(t, y, u) \text{ with } 0 \leq t \leq T \text{ and } u \in \mathcal{U}\}$. The computation of control sets can be reduced to the computation of positive and negative reachable sets, since one easily sees that for a point x in a control set D

$$D = \text{cl}\mathcal{O}^+(x) \cap \mathcal{O}^-(x).$$

Thus a possible strategy to compute a control set is the following: (i) find a point $x \in D$; (ii) compute (an appropriate approximation of) $\text{cl}\mathcal{O}^+(x)$; (iii) compute $\mathcal{O}^-(x)$ and intersect with $\text{cl}\mathcal{O}^+(x)$. The numerical effort in the last step can considerably be reduced by computing only those points in $\mathcal{O}^-(x)$ which are in $\text{cl}\mathcal{O}^+(x)$. For the computation of invariant control sets C which satisfy $C = \text{cl}\mathcal{O}^+(x)$ for all $x \in C$, only step (i) is needed. These sets are particularly relevant, since they determine, under appropriate assumptions, the supports of the invariant measures for an associated stochastic system obtained by formally replacing $u(t)$ by a random perturbation; cp. (Colonius and Kliemann, 1999).

In the computations one has to restrict attention to an a priori given compact subset Q of the state space, and make all computations relative to Q .

2. CONTINUATION WITH DATA REDUCTION

In this section we sketch an algorithm for computing reachable sets which is due to (Häckl, 1996); cp. also (Colonius and Kliemann, 2000). It is based on the observation, that reachable sets are invariant, i.e., for all $y \in \mathcal{O}^+(x)$ and $u \in \mathcal{U}, t > 0$ one has $\varphi(t, y, u) \in \mathcal{O}^+(x)$. The algorithm approximates this invariant set by approximately invariant sets. We define the relative reachable set $\mathcal{O}_Q^+(x)$ as the set of all points which can be reached without leaving Q . In a first step, we consider an associated discrete time system given by a stepsize $\tau > 0$ and

$$x_{k+1} = \varphi(\tau, x_k, u_k), \quad u_k \in \mathcal{U}, \quad k \in \mathbb{Z}. \quad (2.1)$$

Thus in the intervals $(k\tau, (k+1)\tau]$, $k \in \mathbb{Z}$, the controls are constant. Then, for the reachable set $\mathcal{O}_Q^+(\tau, x)$ of these discrete time systems, one can show that the Hausdorff distance $d(\text{cl}\mathcal{O}_Q^+(\tau, x), \text{cl}\mathcal{O}_Q^+(x))$ tends to zero for $\tau \rightarrow 0$.

Definition 2. A set $A \subset Q$ is called ε -invariant with respect to the discrete-time system (2.1) $_\tau$ if $d(A, \varphi(\tau, x, u)) < \varepsilon$ for all $u \in \mathcal{U}$ and all $x \in A$.

The next result relates ε -invariant sets to the orbits of (1.1) and (2.1) $_\tau$.

Proposition 1. Let $I \subset Q$ be an invariant set of (1.1). Fix $x \in I$ and $\tau > 0$. Consider a family $\{I_n, n \in \mathbb{N}\}$ of subsets of Q such that (i) I_n is ε_n -invariant with respect to (2.1) $_\tau$; (ii) $\lim_{n \rightarrow \infty} \varepsilon_n = 0$; (iii) $I_n \subset I_{n+1} \subset I$ for all $n \in \mathbb{N}$; and (iv) there exists $n \in \mathbb{N}$ with $x \in I_n$. Then we have

$$\text{cl}\mathcal{O}^+(\tau, x) \subset \text{cl} \bigcup_{n \in \mathbb{N}} I_n \subset \text{cl} I.$$

Thus it is sufficient to compute ε -invariant sets. Space discretization is obtained in the following way: Let $Q = [a_1, b_1] \times \dots \times [a_d, b_d] \subset \mathbb{R}^d$ be a d -dimensional rectangle. We divide Q into m^d rectangular boxes for some $m \in \mathbb{N}$ by spacing the grid in the i th component as $\delta_i = (b_i - a_i)/m$. Each point $y \in Q$ belongs to exactly one box, denoted by $C(y)$. We denote the grid on Q by

$$\mathcal{G} = \{C_K, K \in \{1, \dots, m\}^d\}, \quad \delta = \max_i \{\delta_i\}. \quad (2.2)$$

Here $K \in \mathbb{Z}^d$ is a multiindex. Let $1 \in \mathbb{Z}^d$ be the index $(1, \dots, 1)$. For two indices $K, N \in \mathbb{Z}^d$ we define $K \pm N = (k_1 \pm n_1, \dots, k_d \pm n_d)$ and $K \leq N$ holds if $k_i \leq n_i$ for all i . For $K \leq N$ the interval $[K, N]$ consists of all indices J with $k_i \leq j_i \leq n_i$ for all i . In particular the neighbors of a box C_K are given by all C_J with $J \in [K-1, K+1]$. Finally, we define for a subset $M \subset Q$ the local convex hull as

$$\text{lco}(M) = \bigcup_{K \in \{1, \dots, m\}^d} \text{co} \left\{ \begin{array}{l} y \in M, C(y) \\ \subset \bigcup_{J \in [K-1, K+1]} C_J \end{array} \right\}.$$

Throughout this section the control range \mathcal{U} is assumed to be the convex hull of finitely many points, $\mathcal{U} = \text{co} \Omega$, $\Omega = \{\omega_1, \dots, \omega_k\} \subset \mathbb{R}^m$.

For the computation of ε -invariant sets we proceed recursively as follows:

Algorithm 1. (Continuation with Data Reduction) Given discretization of Q as in (2.2), a time step $\tau > 0$, and a real parameter $\beta > 0$, define

$$M_0 = \{x_0\} \subset Q, \quad T_0 = \{x_0\} \times \Omega.$$

For $M_k \subset Q$ we set for $(x, \omega) \in T_k$

$$I_k(x, \omega) = \begin{cases} \emptyset & \text{if } d(\varphi(t, x, \omega), \text{lco}(M_k)) < \beta \\ \text{for all } t \in [0, \tau] \\ \{y\} & \text{otherwise,} \end{cases}$$

where in the second case $y = \varphi(t_y, x, \omega)$ for the minimal time $t_y \in [0, \tau]$ such that

$$d(\varphi(t_y, x, \omega), \text{lco}(M_k)) = \beta;$$

$$T_{k+1} = (T_k \setminus \{(x, \omega)\}) \cup (I_k(x, \omega) \times \Omega),$$

$$M_{k+1} = \begin{cases} M_k \cup I_k(x, \omega) & \text{if } T_k \neq \emptyset \\ M_k & \text{if } T_k = \emptyset. \end{cases} \quad (2.3)$$

Theorem 1. For $x \in Q$, Algorithm 1 with

$$\tau_n < 1/n, \beta_n < 1/n, \delta_n = \max_i \{\delta_{in}\} < 1/n$$

leads (after finitely many steps) to an approximating set $M^n \subset \mathcal{O}_Q^+(x)$ such that

$$\lim_{n \rightarrow \infty} d(M^n, \text{cl } \mathcal{O}_Q^+(x)) = 0.$$

The sets M^n , which approximate the reachable set $\mathcal{O}_Q^+(x)$ are all finite, while the orbits $\mathcal{O}_Q^+(x)$ for $x \in \text{int } Q$ have nonvoid interior. Therefore, the local convex hull $\text{lco } M^n$ is a more appropriate approximation of $\mathcal{O}_Q^+(x)$. A slight modification of the result above shows convergence also in this context. Details of an implementation and improvements of the efficiency can be found in (Häckl, 1996). This algorithm gives excellent results in dimension $d = 2$. Also some three dimensional examples have been computed. Here, however, computing times are prohibitive. Computation of local convex hulls also is a critical point in higher dimensions. The computed points have to be stored as floating point numbers; on the other hand, this and the use of local convex hulls for data reduction, makes it possible to use a relatively coarse grid. A definite disadvantage is that recursive refinements of the grid do not seem to be feasible. Nevertheless, it is our feeling that improvements for this algorithm are possible which might make it competitive to others.

3. BOX CONTINUATION

The box continuation algorithm computes relative chain reachable sets and relative chain control sets. For the latter purpose one also has to start with an a priori given point in the relative chain control set. This algorithm has been used in the software package GAIO due to Junge in order to obtain a covering of invariant manifolds of dynamical systems. The discussion here is based on (Szolnoki, 2001). We introduce the following notion of relative chain reachable sets, where the chains have equidistant jump times of length $\tau > 0$. Fix $x, y \in M$ and let $\varepsilon, T > 0$. A *controlled* (ε, T) -chain ζ from x to y is given by $n \in \mathbb{N}$, $x_0, \dots, x_n \in M$, $u_0, \dots, u_{n-1} \in \mathcal{U}$ and jump times $t_0, \dots, t_{n-1} \geq T$ with $x_0 = x$, $x_n = y$ and

$$d(\varphi(t_j, x_j, u_j), x_{j+1}) \leq \varepsilon \text{ for all } j = 0, \dots, n-1.$$

The *positive τ -chain reachable set* of $x \in Q$ relative to Q is defined by

$$\mathcal{O}_{Q, \tau}^+(x) = \left\{ y \in Q, \text{ for all } \varepsilon > 0 \text{ there is a controlled } (\varepsilon, \tau)\text{-chain from } x \text{ to } y \text{ in } Q \text{ with jump times } t_j = \tau \text{ for all } j \right\}.$$

Recall that an \mathbb{R} -viability domain is a subset V of the state space such that for each of its elements there exists a control $u \in \mathcal{U}$ with $\varphi(t, x, u) \in V$ for all $t \in \mathbb{R}$.

Definition 3. A subset E_Q of a closed subset of $Q \subset \mathbb{R}^d$ is called a *chain control set relative to Q* , if (i) E_Q is an \mathbb{R} -viability domain, (ii) for all $x, y \in E_Q$ and $\varepsilon, T > 0$ there is a controlled (ε, T) -chain from x to y , which is completely contained in Q , and (iii) E_Q is maximal (with respect to set inclusion) with these properties.

These relative chain control sets often coincide with control sets. The next result gives an equivalent representation.

Theorem 2. Let Q be a closed subset of M , $E_Q \subset Q$ a chain control set relative to Q and $x \in E_Q$. Then

$$E_Q = \text{Viab}^{\mathbb{R}}(\mathcal{O}_{Q, \tau}^+(x) \cap \mathcal{O}_{Q, \tau}^-(x)).$$

This is the key theorem for the formulation of the algorithm for computation of a chain control set E_Q relative to Q . One starts with a point x in E_Q and computes a suitable covering of its reachable set with the following box continuation algorithm. Fix $\delta > 0$ and consider a partition Q_δ of Q , $Q = \bigcup Q_\delta$, with $\text{diam } Q_\delta < \delta$. Here $\bigcup Q_\delta$ denotes the union of all partition elements contained in Q_δ . For a given $\varepsilon > 0$ we define the *finite positive reachable set* $\mathcal{O}_{Q, \tau, \delta, \varepsilon}^+(x)$ relative to Q of a given point $x \in Q$ by the following box continuation algorithm.

Algorithm 2. (Box Continuation) We define an iteration of continuation steps as follows.

Step 0: Set

$$\Delta_{\varepsilon, 0} := \{B \in Q_\delta, x \in B\}, \mathcal{O}_{\varepsilon, 0}^+(x) := \Delta_{\varepsilon, 0}.$$

Step k : ($k = 1, 2, 3, \dots$) At the beginning of selection step k we have a collection $\mathcal{O}_{\varepsilon, k-1}^+(x)$.

Define a new collection $\mathcal{O}_{\varepsilon, k}^+(x)$ by

$$\Delta_{\varepsilon, k} = \left\{ B \in Q_\delta, \text{ there exist } \hat{B} \in \Delta_{\varepsilon, k-1}, \right. \\ \left. u \in \mathcal{U}, \text{ such that } B \in \Phi_\tau^\varepsilon(\hat{B}, u) \right\}$$

$$\mathcal{O}_{\varepsilon, k}^+(x) = \mathcal{O}_{\varepsilon, k-1}^+(x) \cup \Delta_{\varepsilon, k},$$

where

$$\Phi_\tau^\varepsilon(\hat{B}, u) := \left\{ B \in Q_\delta, B \cap \bigcup_{x \in \varphi_\tau(\hat{B}, u)} B_\varepsilon(x) \neq \emptyset \right\}$$

The algorithm terminates if $\Delta_{\varepsilon, \ell} = \emptyset$, where $\ell \leq |Q_\delta|$ and $|Q_\delta|$ denotes the number of elements in Q_δ and we set

$$\mathcal{O}_{Q, \tau, \delta, \varepsilon}^+(x) := \mathcal{O}_{\varepsilon, \ell}^+(x).$$

Hence in every step of the box continuation algorithm, we map each element of Q_δ added in the

previous continuation step, enlarge the image by an ε -neighborhood, and add all those elements of Q_δ having a nonvoid intersection with this enlarged image. Then the following convergence result holds, see (Szolnoki, 2001).

Theorem 3. Let $Q \subset \mathbb{R}^d$ and consider strictly decreasing sequences $\delta_i \rightarrow 0^+$, $\varepsilon_i \rightarrow 0^+$ for $i \rightarrow \infty$ and a corresponding sequence of successively finer partitions Q_{δ_i} of Q with $\text{diam}(Q_{\delta_i}) \leq \delta_i$ for all $i \in \mathbb{N}_0$. Then the finite positive reachable sets $\mathcal{O}_{Q,\tau,\delta_i,\varepsilon_i}^+(x)$ define a decreasing sequence with

$$\mathcal{O}_{Q,\tau}^+(x) = \bigcap_{j=0}^{\infty} \mathcal{O}_{Q,\tau,\delta_j,\varepsilon_j}^+(x).$$

Note that the enlargement of the box images by ε -neighborhoods are imposed for technical reasons, but not used in the current implementation. In other words, in practice we use Algorithm 2 with $\varepsilon = 0$. In all numerical examples this was sufficient. Furthermore, in Theorem 2, computation of the viability kernel of the intersection is not necessary, if the intersection is viable (this, usually, is the case). Once the initial point is known, it can be used to obtain a covering of the control set. The actual implementation has to deal with the problem, that the image $\Phi_\tau^\varepsilon(\tilde{B}, u)$ of a box \tilde{B} under a control u cannot precisely be computed. Hence it has to be replaced by the image under a number of test points. Also measurable controls are not implementable; thus also the controls are replaced by a finite number of constant controls, distributed according to different criteria over the control range U . The box continuation algorithm suffers, as the preceding one, from the deficiency that successive refinements of the grid do not appear feasible. On the other hand, continuation algorithms in particular provide information on the time needed to reach a specified point. This is in contrast to subdivision methods described below. Numerical experience shows that the box continuation algorithm tends to overestimate reachable sets as is to be expected from the monotonicity property in Theorem 3.

4. SUBDIVISION ALGORITHM FOR VIABILITY KERNELS

The subdivision algorithm is based on an approach by (Dellnitz and Hohmann, 1997) and (Dellnitz and Junge, 1999) for computation of relative attractors in dynamical systems. An implementation based on GAIO (Junge) is given in (Szolnoki, 2001). It computes relative attractors which, in the context of control systems, are viability kernels. It can be used to compute reachable sets and also relative chain control sets

and their domains of attraction, based on the relation between viability kernels and control sets, see (Szolnoki, 2000). First we discretize time and obtain the corresponding exact time discretization for time step $\tau > 0$

$$\begin{aligned} x_{k+1} &= \varphi(\tau, x_k, u(t_k + \cdot)), \quad u \in U, \\ t_0 &= 0, \quad t_{k+1} = t_k + \tau, \quad k \in \mathbb{Z}. \end{aligned} \quad (4.1)$$

The unique solution for $x_0 \in \mathbb{R}^d$ and $u \in U$ is denoted by $(x_k)_{k \in \mathbb{Z}}^u$.

Definition 4. The *discrete viability kernel* of a given closed set $Q \subset \mathbb{R}^d$ is defined by

$$\text{Viab}_\tau(Q) := \left\{ x_0 \in Q, \text{ there exists } u \in U \text{ such that } (x_k)_{k \in \mathbb{Z}}^u \subset Q \right\}.$$

We suppose that $\text{Viab}_\tau(Q)$ lies sufficiently “deep” inside Q , that is,

$$B_{\frac{1}{2}K\tau}(x_0) \subset Q \quad \text{for all } x_0 \in \text{Viab}_\tau(Q), \quad (4.2)$$

where K denotes the bound value of the right hand side of (1.1).

Proposition 2. Let Q be a compact subset of \mathbb{R}^d and suppose that assumption (4.2) is satisfied. Then

$$\text{Viab}_\tau(Q) = \text{Viab}^{\mathbb{R}}(Q).$$

For the rest of this section we assume that assumption (4.2) is satisfied. Hence we may restrict our attention to discrete viability kernels. We now describe an algorithm for the approximation of $\text{Viab}_\tau(Q)$. Consider a partition Q_δ of Q with $\text{diam}(Q_\delta) \leq \delta$. For $B \in Q_\delta$, we denote

$$\begin{aligned} \varphi_\tau(B, u) &:= \left\{ y \in Q, \text{ there exists } x \in B \text{ such that } y = \varphi(\tau, x, u) \right\}, \\ \Phi_\tau(B, u) &:= \{ \tilde{B} \in Q_\delta, \varphi_\tau(B, u) \cap \tilde{B} \neq \emptyset \} \end{aligned}$$

and define a dynamic on Q_δ , which is in fact a difference inclusion:

$$\begin{aligned} B_{k+1} &\in \Phi_\tau(B_k, u(t_k + \cdot)), \quad u \in U, \\ t_0 &= 0, \quad t_{k+1} = t_k + \tau, \quad k \in \mathbb{Z}. \end{aligned} \quad (4.3)$$

For $B_0 \in Q_\delta$ and $u \in U$, the (nonunique) solution of (4.3) is denoted by $(B_k)_{k \in \mathbb{Z}}^u$.

The finite viability kernel $\text{Viab}_{\tau,\delta}(Q_\delta)$ given by

$$\text{Viab}_{\tau,\delta}(Q_\delta) := \left\{ B_0 \in Q_\delta, \text{ there exists } u \in U \text{ such that } (B_k)_{k \in \mathbb{Z}}^u \subset Q_\delta \right\}.$$

can be computed in a finite number of steps by the following algorithm.

Algorithm 3. (Finite viability kernels) We define an iteration of selection steps as follows.

Step 0: Set $Q^0(Q_\delta) := Q_\delta$.

Step k: ($k = 1, 2, 3, \dots$) At the beginning of selection step k we have a collection $\mathcal{Q}^{k-1}(Q_\delta)$. Define a new collection $\mathcal{Q}^k(Q_\delta)$ by

$$\mathcal{Q}^k(Q_\delta) := \left\{ \begin{array}{l} B \in \mathcal{Q}^{k-1}(Q_\delta), \text{ there exist} \\ u_1, u_2 \in \mathcal{U} \text{ such that} \\ \Phi_\tau(B, u_1) \cap \mathcal{Q}^{k-1}(Q_\delta) \neq \emptyset \text{ and} \\ \Phi_{-\tau}(B, u_2) \cap \mathcal{Q}^{k-1}(Q_\delta) \neq \emptyset \end{array} \right\}.$$

The algorithm terminates if $\mathcal{Q}^\ell(Q_\delta) = \mathcal{Q}^{\ell-1}(Q_\delta)$, where $\ell \leq |Q_\delta|$ and $|Q_\delta|$ denotes the number of elements in Q_δ .

Assume that $Q = \bigcup Q_{\delta_i}$ for each $i \in \mathbb{N}_0$ and that for any $B_{\delta_i} \in Q_{\delta_i}$, there exists $B_{\delta_{i-1}} \in Q_{\delta_{i-1}}$ such that B_{δ_i} is contained in $B_{\delta_{i-1}}$, except for a null set and for every $B_{\delta_i} \in Q_{\delta_i}$, there exists $B_{\delta_{i-1}} \in Q_{\delta_{i-1}}$ such that $B_{\delta_i} \subset B_{\delta_{i-1}}$. Note that for partition refinements obtained by a subdivision method these assumptions are fulfilled.

Algorithm 4. (Subdivision algorithm for viability kernels) For a given strictly decreasing sequence $\delta_i \rightarrow 0^+$ consider a corresponding sequence of successively finer partitions Q_{δ_i} of a closed set Q with $\text{diam}(Q_{\delta_i}) \leq \delta_i$ for all $i \in \mathbb{N}_0$. Choose a sequence $(k_i)_{i \in \mathbb{N}}$, $k_i \in \mathbb{N}$ for all $i \in \mathbb{N}$.

Step 0 Set $\mathcal{W}_0 := Q_{\delta_0} \subset \Psi_{\delta_0}$.

Step i ($i = 1, 2, 3, \dots$) At the beginning of step i we have a collection $\mathcal{W}_{i-1} \subset Q_{\delta_{i-1}}$. We obtain a new collection $\mathcal{W}_i \subset Q_{\delta_i}$ in two steps.

i.1 (Subdivision step) Change to partition Q_{δ_i} by defining a collection $\hat{\mathcal{W}}_i \subset Q_{\delta_i}$ such that

$$U\hat{\mathcal{W}}_i = U\mathcal{W}_{i-1}.$$

i.2 (Selection step) Compute a new collection $\mathcal{W}_i \subset Q_{\delta_i}$ with k_i steps of algorithm 3, that is,

$$\mathcal{W}_i := \mathcal{Q}^{k_i}(\hat{\mathcal{W}}_i).$$

Convergence is guaranteed by the following theorem.

Theorem 4. Let Q be a compact subset of \mathbb{R}^d and let $\mathcal{W}_0, \mathcal{W}_1, \dots$ be the sequence of collections generated by Algorithm 4. Then we obtain a decreasing sequence of collections $\mathcal{W}_0, \mathcal{W}_1, \dots$ with $U\mathcal{W}_i \subset U\mathcal{W}_{i-1}$ for all $i \in \mathbb{N}$ and

$$\text{Viab}_\tau(Q) = \bigcap_{i=0}^{\infty} U\mathcal{W}_i.$$

A major advantage of the subdivision algorithm is that the grid can be stored via a tree data structure: Then also iterative refinements are possible and rather fine grids can be handled.

Remark 1. (Grüne, 2001) describes rigorous space discretizations ensuring convergence; furthermore, he introduces an adaptive subdivision procedure

ensuring that asymptotically only those boxes are subdivided which describe the boundary of the domain of attraction. Thus, essentially, the problem dimension is diminished by one.

5. SUBDIVISION ALGORITHM FOR SYMBOLIC IMAGES

If one is interested in computing chain control sets, an alternative subdivision procedure can be performed directly on an associated directed graph, whose vertices are the partition elements (boxes); an edge connects two vertices if there are a point in the first box and a control steering the system to the other box. Then one computes the strongly connected components of the graph. The original idea, apparently, is due to (Osipenko, 1983) who called the resulting graph the symbolic image of the dynamical system. When combined with subdivision techniques this becomes a powerful tool. In GAIO it is used for computing chain recurrent components for dynamical systems, compare (Dellnitz *et al.*, 2000). This has been extended by (Szolnoki, 2001) for control systems, as described below.

Algorithm 5. (Subdivision or symbolic images) For a closed subset Q and strictly decreasing sequences $\delta_i \rightarrow 0^+$ and $\varepsilon_i \rightarrow 0^+$ consider a corresponding sequence of successively finer partitions Q_{δ_i} with $\text{diam}(Q_{\delta_i}) \leq \delta_i$ for all $i \in \mathbb{N}_0$.

Step 0 Set $S_0 := Q_{\delta_0} \subset \Psi_{\delta_0}$.

Step i ($i = 1, 2, 3, \dots$) At the beginning of step i we have a collection $S_{i-1} \subset Q_{\delta_{i-1}}$. We obtain a new collection $S_i \subset Q_{\delta_i}$ in two steps.

i.1 (Subdivision step) Change to partition Q_{δ_i} by defining a collection $\hat{S}_i \subset Q_{\delta_i}$ such that

$$U\hat{S}_i = US_{i-1}.$$

i.2 (Selection step) Construct a directed graph $G_i = (V_i, E_i)$, where the vertices V and the edges E are given by

$$V_i = \hat{S}_i \text{ and}$$

$$E_i = \left\{ (B, B') \in \hat{S}_i \times \hat{S}_i, B' \in \Phi_\tau^{\varepsilon_i}(B, u) \right\} \text{ for some } u \in \mathcal{U}$$

where $\Phi_\tau^{\varepsilon_i}(B, u)$ is the set of all $\hat{B} \in Q_\delta$ with $\hat{B} \cap \bigcup_{x \in \varphi_\tau(B, u)} B_{\varepsilon_i}(x) \neq \emptyset$. Compute the strongly connected components $S_{i,1}, \dots, S_{i,r_i}$ of G_i , $r_i \in \mathbb{N}$, and set

$$S_i = \{ B \in \hat{S}_i, B \in S_{i,k} \text{ for some } k \}.$$

By construction, the algorithm generates a sequence S_0, S_1, \dots of collections with $US_i \subset US_{i-1}$ for all $i \in \mathbb{N}$.

Theorem 5. For Algorithm 5

$$\text{Viab}^{\mathbb{R}}(S_\infty) = \bigcup E_Q,$$

where $\mathcal{S}_\infty := \bigcap_{i=0}^\infty U\mathcal{S}_i$, and the union is taken over all relative chain control sets E_Q in Q .

Compared to the first subdivision algorithm, the present algorithm requires considerably more memory for the computation of the strongly connected components. To some degree this is also true for nonrecursive implementations of the algorithm for computing strongly connected components. On the other hand, these graph algorithms are much faster than numerical solutions of ordinary differential equations.

6. DISCUSSION

A fundamental advantage of the subdivision algorithm, Algorithm 5, is that it can detect all chain control sets. Contrary to the other algorithms, it is not necessary to know a priori a point in the chain control set. This is particularly important when there exist chain control sets which do not contain a limit set of the uncontrolled system (e.g., this seems to be the case in the so-called escape equation, a simplified model for ship roll motion under wave forcing, see (Fischer *et al.*, 2000)). In view of the larger memory requirements of this algorithm the following combination of algorithms has proved reasonable: First, the strongly connected components are computed with Algorithm 5 to obtain a suitable covering of all relative chain control sets. Then we switch to Algorithm 4 and approximate the viability kernel of the strongly connected components as an approximation of $\bigcup_{E_Q} E_Q$. Thus, once the control sets are localized with the graph algorithm on a comparatively coarse partition, we can switch to the “cheap” subdivision algorithm. Similarly, one can also combine Algorithms 2 and 4.

7. REFERENCES

- Camilli, F., L. Grüne and F. Wirth (2000). A regularization of Zubov's equation for robust domains of attraction. In: *Nonlinear Control in the Year 2000* (A. Isidori, F. Lamnabhi-Lagarrigue and W. Respondek, Eds.), pp. 277–290. Springer-Verlag.
- Colonius, F. and W. Kliemann (1999). Continuous, smooth and control techniques for stochastic dynamics. In: *Stochastic Dynamics* (H. Crauel and M. Gundlach, Eds.). Springer-Verlag.
- Colonius, F. and W. Kliemann (2000). *The Dynamics of Control*. Birkhäuser.
- Dang, T. and O. Maler (1998). Reachability Analysis via Face Lifting. In: *Hybrid Systems: Computation and Control* (T.A. Henzinger and S. Sastry, Eds.), Springer-Verlag. pp. 96–109.
- Dellnitz, M. and A. Hohmann (1997). A Subdivision Algorithm for the Computation of Unstable Manifolds and Global Attractors. *Numerische Mathematik* **75**, 293–317.
- Dellnitz, M. and O. Junge (1999). On the Approximation of Complicated Dynamical Behavior. *SIAM J. Numer. Anal.* **36**, 491–515.
- Dellnitz, M., O. Junge, M. Rumpf and R. Strzodka (2000). The computation of an unstable invariant set inside a cylinder containing a knotted flow. In: *Proc. Int. Conf. Differential Equations, Berlin 1999* (B. Fiedler, K. Gröger and J. Sprekels, Eds.). World Scientific. pp. 1053–1059.
- Falcone, M., L. Grüne and F. Wirth (2000). A maximum time approach to the computation of robust domains of attraction. In: *Proc. Int. Conf. Differential Equations, Berlin 1999* (B. Fiedler, K. Gröger and J. Sprekels, Eds.). World Scientific. pp. 844–849.
- Fischer, J., R. Guder and E. Kreuzer (2000). Analyzing Perturbed Nonlinear Dynamical Systems. In: *Proc. 9th German-Japanese Seminar "Nonlinear Problems in Dynamical Systems"*, Straelen, Germany. To appear.
- Grüne, L. (2001). Asymptotic Behavior of Dynamical and Control Systems under Perturbation and Discretization. to appear.
- Häckl, G. (1996). Reachable Sets, Control Sets and Their Computation. Dissertation, Universität Augsburg, Augsburger Mathematische Schriften, Band 7.
- Kurzhanski, A. and I. Valyi (1998). *Ellipsoidal Calculus for Estimation and Control*. Birkhäuser.
- Osipenko, G. (1983). On a Symbolic Image of Dynamical System. In: *Boundary value problems, Perm.* pp. 101–105. In Russian.
- Szolnoki, D. (2000). Viability Kernels and Control Sets. *ESAIM Control, Optimisation and Calculus of Variations* **5**, 175–185.
- Szolnoki, D. (2001). Algorithms for Reachability problems. Dissertation, Institut für Mathematik, Universität Augsburg, Augsburg.